Notes on Blockchain Governance

In which I argue that "tightly coupled" on-chain voting is overrated, the status quo of "informal governance" as practiced by Bitcoin, Bitcoin Cash, Ethereum, Zcash and similar systems is much less bad than commonly thought, that people who think that the purpose of blockchains is to completely expunge soft mushy human intuitions and feelings in favor of completely algorithmic governance (emphasis on "completely") are absolutely crazy, and loosely coupled voting as done by Carbonvotes and similar systems is underrated, as well as describe what framework should be used when thinking about blockchain governance in the first place.

See also: [https://medium.com/@Vlad_Zamfir/against-on-chain-governance-a4ceacd040ca

](https://medium.com/@Vlad_Zamfir/against-on-chain-governance-a4ceacd040ca)

One of the more interesting recent trends in blockchain governance is the resurgence of on-chain coin-holder voting as a multi-purpose decision mechanism. Votes by coin holders are sometimes used in order to decide who operates the super-nodes that run a network (eg. DPOS in EOS, NEO, Lisk and other systems), sometimes to vote on protocol parameters (eg. the Ethereum gas limit) and sometimes to vote on and directly implement protocol upgrades wholesale (eg. Tezos). In all of these cases, the votes are automatic - the protocol itself contains all of the logic needed to change the validator set or to update its own rules, and does this automatically in response to the result of votes.

Explicit on-chain governance is typically touted as having several major advantages. First, unlike the highly conservative philosophy espoused by Bitcoin, it can evolve rapidly and accept needed technical improvements. Second, by creating an explicit

decentralized framework, it avoids the perceived pitfalls of informal

governance, which is viewed to either be too unstable and prone to chain splits, or prone to becoming too de-facto centralized - the latter being the same argument made in the famous 1972 essay "Tyranny of Structurelessness".

Quoting Tezos documentation:

While all blockchains offer financial incentives for maintaining consensus on their ledgers, no blockchain has a robust on-chain mechanism that seamlessly amends the rules governing its protocol and rewards protocol development. As a result, first-generation blockchains empower de facto, centralized core development teams or miners to formulate design choices.

And:

Yes, but why would you want to make [a minority chain split] easier? Splits destroy network effects.

On-chain governance used to select validators also has the benefit that it allows for networks that impose high computational performance requirements on validators without introducing economic centralization risks and other traps of the kind that appear in public blockchains (eg. the validator's dilemma).

So far, all in all, on-chain governance seems like a very good bargain.... so what's wrong with it?

What is Blockchain

Governance?

To start off, we need to describe more clearly what the process of "blockchain governance" is

. Generally speaking, there are two informal models of governance, that I will call the "decision function" view of governance and the "coordination" view of governance. The decision function view treats governance as a function (f(x_1, x_2 ... x_n) \rightarrow y)

, where the inputs are the wishes of various legitimate stakeholders (senators, the president, property owners, shareholders, voters, etc) and the output is the decision.

The decision function view is often useful as an approximation, but it clearly frays very easily around the edges: people often can and do break the law and get away with it, sometimes rules are ambiguous, and sometimes revolutions happen - and all three of these possibilities are, at least sometimes, a good thing

. And often even behavior inside the system is shaped by incentives created by the possibility

of acting outside the system, and this once again is at least sometimes a good thing.

The coordination model of governance, in contrast, sees governance as something that exists in layers. The bottom layer is, in the real world, the laws of physics themselves (as a geopolitical realist would say, guns and bombs), and in the blockchain space we can abstract a bit further and say that it is each individual's ability to run whatever software they want in their capacity as a user, miner, stakeholder, validator or whatever other kind of agent a blockchain protocol allows them to be.

The bottom layer is always the ultimate deciding layer; if, for example, all Bitcoin users wake up one day and decides to edit their clients' source code and replace the entire code with an Ethereum client that listens to balances of a particular ERC20 token contract, then that means that that ERC20 token is

bitcoin. The bottom layer's ultimate governing power cannot be stopped, but the actions that people take on this layer can be influenced

by the layers above it.

The second (and crucially important) layer is coordination institutions. The purpose of a coordination institution is to create focal points around how and when individuals should act in order to better coordinate behavior. There are many situations, both in blockchain governance and in real life, where if you act in a certain way alone, you are likely to get nowhere (or worse), but if everyone acts together a desired result can be achieved.

In these cases, it's in your interest to go if everyone else is going, and stop if everyone else is stopping. You can think of coordination institutions as putting up green or red flags in the air saying "go" or "stop", with an established culture

that everyone watches these flags and (usually) does what they say. Why do people have the incentive to follow these flags? Because everyone else

is already following these flags, and you have the incentive to do the same thing as what everyone else is doing.

In the real world, military orders from a general function as a flag, and in the blockchain world, the simplest example of such a flag is the mechanism that tells people whether or not a hard fork "is happening". Coordination institutions can be very formal, or they can be informal, and often give suggestions that are ambiguous. Flags would ideally always be either red or green, but sometimes a flag might be yellow, or even holographic, appearing green to some participants and yellow or red to others. Sometimes that are also multiple flags that conflict with each other.

The key questions of governance thus become:

- What should layer 1 be? That is, what features should be set up in the initial protocol itself, and how does this influence the ability to make formulaic (ie. decision-function-like) protocol changes, as well as the level of power of different kinds of agents to act in different ways?
- What should layer 2 be? That is, what coordination institutions should people be encouraged to care about?

The Role of Coin Voting

Ethereum also has a history with coin voting, including:

- DAO proposal votes

: https://daostats.github.io/proposals.html

- The DAO Carbonvote

: http://v1.carbonvote.com/

- The EIP 186/649/669 Carbonvote

: http://carbonvote.com/

These three are all examples of loosely coupled

coin voting, or coin voting as a layer 2 coordination institution. Ethereum does not have any examples of tightly coupled

coin voting (or, coin voting as a layer 1 in-protocol feature), though it does

have an example of tightly coupled miner

voting: miners' right to vote on the gas limit. Clearly, tightly coupled voting and loosely coupled voting are competitors in the governance mechanism space, so it's worth dissecting: what are the advantages and disadvantages of each one?

Assuming zero transaction costs, and if used as a sole governance mechanism, the two are clearly equivalent. If a loosely coupled vote says that change X should be implemented, then that will serve as a "green flag" encouraging everyone to download the update; if a minority wants to rebel, they will simply not download the update. If a tightly coupled vote implements change X, then the change happens automatically, and if a minority wants to rebel they can install a hard fork update that cancels the change. However, there clearly are nonzero transaction costs associated with making a hard fork, and this leads to some very important differences.

One very simple, and important, difference is that tightly coupled voting creates a default in favor of the blockchain adopting what the majority wants, requiring minorities to exert great effort to coordinate a hard fork to preserve a blockchain's existing

properties, whereas loosely coupled voting is only a coordination tool, and still requires users to actually download and run the software that implements any given fork. But there are also many other differences. Now, let us go through some arguments against

voting, and dissect how each argument applies to voting as layer 1 and voting as layer 2.

Low Voter Participation

One of the main criticisms of coin voting mechanisms so far is that, no matter where they are tried, they tend to have very low voter participation. The DAO Carbonvote only had a voter participation rate of 4.5%:

Additionally, wealth distribution is very unequal, and the results of these two factors together are best described by this image created by a critic of the DAO fork:

The EIP 186 Carbonvote had ~2.7 million ETH voting. The DAO proposal votesdid not fare better, with participation never reaching 10%. And outside of Ethereum things are not sunny either; even in Bitshares, a system where the core social contract is designed around voting, the top delegate in an approval vote only got 17% of the vote, and in Lisk it gotup to 30%, though as we will discuss later these systems have other problems of their own.

Low voter participation means two things. First, the vote has a harder time achieving a perception of legitimacy, because it only reflects the views of a small percentage of people. Second, an attacker with only a small percentage of all coins can sway the vote. These problems exist regardless of whether the vote is tightly coupled or loosely coupled.

Game-Theoretic Attacks

Aside from "the big hack" that received the bulk of the media attention, the DAO also had a number of much smaller game-theoretic vulnerabilities; this article from HackingDistributed does a good job of summarizing them. But this is only the tip of the iceberg. Even if all of the finer details of a voting mechanism are implemented correctly, voting mechanisms in general have a large flaw: in any vote, the probability that any given voter will have an impact on the result is tiny, and so the personal incentive that each voter has to vote correctly is almost insignificant. And if each person's size of the stake is small, their incentive to vote correctly is insignificant squared

. Hence, a relatively small bribe spread out across the participants may suffice to sway their decision, possibly in a way that they collectively might quite disapprove of.

Now you might say, people are not evil selfish profit-maximizers that will accept a $0.5 bribe to vote to give twenty million dollars to Josh arza just because the above calculation says their individual chance of affecting anything is tiny; rather, they would altruistically refuse to do something that evil. There are two responses to this criticism.

First, there are ways to make a "bribe" that are quite plausible; for example, an exchange can offer interest rates for deposits (or, even more ambiguously, use the exchange's own money to build a great interface and features), with the exchange operator using the large quantity of deposits to vote as they wish. Exchanges profit from chaos, so their incentives are clearly quite misaligned with users and

coin holders.

Second, and more damningly, in practice it seems like people, at least in their capacity as crypto token holders, are

profit maximizers, and seem to see nothing evil or selfish about taking a bribe or two. As "Exhibit A", we can look at the situation with Lisk, where the delegate pool seems to have been successfully captured by two major "political parties" that explicitly bribe coin holders to vote for them, and also require each member in the pool to vote for all the others.

Here's LiskElite, with 55 members (out of a total 101):

Here's LiskGDT, with 33 members:

And as "Exhibit B" some voter bribes being paid outin Ark:

Here, note that there is a key difference between tightly coupled and loosely coupled votes. In a loosely coupled vote, direct or indirect vote bribing is also possible, but if the community agrees that some given proposal or set of votes constitutes a game-theoretic attack, they can simply socially agree to ignore it. And in fact this has kind of already happened - the Carbonvote contains a blacklist of addresses corresponding to known exchange addresses, and votes from these addresses are not counted. In a tightly coupled vote, there is no way to create such a blacklist at protocol level, because agreeing who is part of the blacklist is itself

a blockchain governance decision. But since the blacklist is part of a community-created voting tool that only indirectly influences protocol changes, voting tools that contain bad blacklists can simply be rejected by the community.

It's worth noting that this section is not

a prediction that all tightly coupled voting systems will quickly succumb to bribe attacks. It's entirely possible that many will

survive for one simple reason: all of these projects have founders or foundations with large premines, and these act as large centralized actors that are interested in their platforms' success that are not vulnerable to bribes, and hold enough coins to outweigh most bribe attacks. However, this kind of centralized trust model, while arguably useful in some contexts in a project's early stages, is clearly one that is not sustainable in the long term.

Non-Representativeness

Another important objection to voting is that coin holders are only one class of user, and may have interests that collide with those of other users. In the case of pure cryptocurrencies like Bitcoin, store-of-value use ("hodling") and medium-of-exchange use ("buying coffees") are naturally in conflict, as the store-of-value prizes security much more than the medium-of-exchange use case, which more strongly values usability. With Ethereum, the conflict is worse, as there are many people who use Ethereum for reasons that have nothing to do with ether (see: cryptokitties), or even value-bearing digital assets in general (see: ENS).

Additionally, even if coin holders are

the only relevant class of user (one might imagine this to be the case in a cryptocurrency where there is an established social contract that its purpose is to be the next digital gold, and nothing else), there is still the challenge that a coin holder vote gives a much greater voice to wealthy coin holders than to everyone else, opening the door for centralization of holdings to lead to unencumbered centralization of decision making. Or, in other words...

And if you want to see a review of a project that seems to combine all of these disadvantages at the same time, see this: https://btcgeek.com/bitshares-trying-memorycoin-year-ago-disastrous-ends/.

This criticism applies to both tightly coupled and loosely coupled voting equally; however, loosely coupled voting is more amenable to compromises that mitigate its unrepresentativeness, and we will discuss this more later.

Centralization

Let's look at the existing live experiment that we have in tightly coupled voting on Ethereum, the gas limit. Here's the gas limit evolution over the past two years:

You might notice that the general feel of the curve is a bit like another chart that may be quite familiar to you:

Basically, they both look like magic numbers that are created and repeatedly renegotiated by a fairly centralized group of guys sitting together in a room. What's happening in the first case? Miners are generally following the direction favored by the community, which is itself gauged via social consensus aids similar to those that drive hard forks (core developer support, Reddit upvotes, etc; in Ethereum, the gas limit has never gotten controversial enough to require anything as serious as a coin vote).

Hence, it is not at all clear that voting will be able to deliver results

that are actually decentralized, if voters are not technically knowledgeable and simply defer to a single dominant tribe of experts. This criticism once again applies to tightly coupled and loosely coupled voting equally.

Update: since writing this, it seems like Ethereum miners managed to up the gas limit from 6.7 million to 8 million all without even discussing it with the core developers or the Ethereum Foundation. So there is hope; but it takes a lot of hard community building and other grueling non-technical work to get to that point.

Digital Constitutions

One approach that has been suggested to mitigate the risk of runaway bad governance algorithms is "digital constitutions" that mathematically specify desired properties that the protocol should have, and require any new code changes to come with a computer-verifiable proof that they satisfy these properties. This seems like a good idea at first, but this too should, in my opinion, be viewed skeptically.

In general, the idea of having norms about protocol properties, and having these norms serve the function of one of the coordination flags, is a very good one. This allows us to enshrine core properties of a protocol that we consider to be very important and valuable, and make them more difficult to change. However, this is exactly the sort of thing that should be enforced in loosely coupled (ie. layer two), rather than tightly coupled (layer one) form.

Basically any meaningful norm is actually quite hard to express in its entirety; this is part of the complexity of value problem. This is true even for something as seemingly unambiguous as the 21 million coin limit. Sure, one can add a line of code saying assert total_supply <= 21000000

, and put a comment around it saying "do not remove at all costs", but there are plenty of roundabout ways of doing the same thing. For example, one could imagine a soft fork that adds a mandatory transaction fee this is proportional to coin value * time since the coins were last sent, and this is equivalent to demurrage, which is equivalent to deflation. One could also implement another currency, called Bjtcoin, with 21 million new

units, and add a feature where if a bitcoin transaction is sent the miner can intercept it and claim the bitcoin, instead giving the recipient bjtcoin; this would rapidly force bitcoins and bjtcoins to be fungible with each other, increasing the "total supply" to 42 million without ever tripping up that line of code. "Softer" norms like not interfering with application state are even harder to enforce.

We want

to be able to say that a protocol change that violates any of these guarantees should be viewed as illegitimate - there should be a coordination institution that waves a red flag - even if they get approved by a vote. We also want to be able to say that a protocol change that follows the letter of a norm, but blatantly violates its spirit, the protocol change should still

be viewed as illegitimate. And having norms exist on layer 2 - in the minds of humans in the community, rather than in the code of the protocol - best achieves that goal.

Toward A Balance

However, I am also not willing to go the other way and say that coin voting, or other explicit on-chain voting-like schemes, have no place in governance whatsoever. The leading alternative seems to be core developer consensus, however the failure mode of a system being controlled by "ivory tower intellectuals" who care more about abstract philosophies and solutions that sound technically impressive over and above real day-to-day concerns like user experience and transaction fees is, in my view, also a real threat to be taken seriously.

So how do we solve this conundrum? Well, first, we can heedthe words of slatestarcodex in the context of traditional politics:

The rookie mistake is: you see that some system is partly Moloch [ie. captured by misaligned special interests], so you say "Okay, we'll fix that by putting it under the control of this other system. And we'll control this other system by writing ‘DO NOT BECOME MOLOCH' on it in bright red marker." ("I see capitalism sometimes gets misaligned. Let's fix it by putting it under control of the government. We'll control the government by having only virtuous people in high offices.") I'm not going to claim there's a great alternative, but the occasionally-adequate alternative is the neoliberal one – find a couple of elegant systems that all optimize along different criteria approximately aligned with human happiness, pit them off against each other in a structure of checks and balances, hope they screw up in different places like in that swiss cheese model, keep enough individual free choice around that people can exit any system that gets too terrible, and let cultural evolution do the rest.

In blockchain governance, it seems like this is the only way forward as well. The approach for blockchain governance that I advocate is "multifactorial consensus", where different coordination flags and different mechanisms and groups are polled, and the ultimate decision depends on the collective result of all of these mechanisms together. These coordination flags may include:

- The roadmap (ie. the set of ideas broadcasted earlier on in the project's history about the direction the project would be going)
- Consensus among the dominant core development teams
- Coin holder votes
- User votes, through some kind of sybil-resistant polling system
- Established norms (eg. non-interference with applications, the 21 million coin limit)

I would argue that it is very useful for coin voting to be one of several coordination institutions deciding whether or not a given change gets implemented. It is an imperfect and unrepresentative signal, but it is a Sybil-resistant

one - if you see 10 million ETH voting for a given proposal, you cannot

dismiss that by simply saying "oh, that's just hired Russian trolls with fake social media accounts". It is also a signal that is sufficiently disjoint from the core development team that if needed it can serve as a check on it. However, as described above, there are very good reasons why it should not be the only

coordination institution.

And underpinnning it all is the key difference from traditional systems that makes blockchains interesting: the "layer 1" that underpins the whole system is the requirement for individual users to assent to any protocol changes, and their freedom, and credible threat, to "fork off" if someone attempts to force changes on them that they consider hostile (see also: http://vitalik.ca/general/2017/05/08/coordination_problems.html).

Tightly coupled voting is also okay to have in some limited contexts - for example, despite its flaws, miners' ability to vote on the gas limit is a feature that has proven very beneficial on multiple occasions. The risk that miners will try to abuse their power may well be lower than the risk that any specific gas limit or block size limit hard-coded by the protocol on day 1 will end up leading to serious problems, and in that case letting miners vote on the gas limit is a good thing. However, "allowing miners or validators to vote on a few specific parameters that need to be rapidly changed from time to time" is a very far cry

from giving them arbitrary control over protocol rules, or letting voting control validation, and these more expansive visions of on-chain governance have a much murkier potential, both in theory and in practice.